Appendix A

Stage II Precipitation Processing System

User's Guide

Stage II Precipitation Processing System

User's Guide

NWS Office of Hydrology
Hydrologic Research Laboratory

July 1997

Table of Contents

I. Introduction

This document contains system information such as directory and file information in addition to user information such as how to run Stage II manually. In its current form, Stage II is a non-interactive process. It can be run automatically from a crontab, manually from the command line or from within the Stage III process.

A. Overview of Stage II Process

The Stage II process takes radar input from the WSR-88D RPG running Stage I and gage data from the SIIPP (Stage II preprocessor) process to generate a gage-only field and a multisensor field. These fields are currently output to flat files for use by Stage III. Future enhancements will store these outputs in the Informix database.

Stage II processing consists of 5 algorithms executed in the following order:

(1) quality control
(2) time distribution of multi-hour gage data
(3) summary statistics calculation
(4) gage-only field generation
(5) multisensor field generation

See Section V for further information on each algorithm.

B. Description of Development Environment

Stage II was originally written on a PRIME 9755 computer at OH as part of the PROTEUS project. The code was all Fortran 77 including embedded SQL statements for accessing the RAPPORT database. The code was then moved to an IBM RS/6000 workstation with the database interactions replaced by flat file reads/writes. When the Informix DBMS was installed, the flat file input/output was replaced by database interactions using ESQL/FORTRAN. When the AWIPS requirements were defined, the database interactions were rewritten using ESQL/C. The code was then ported to the Government Development Platforms (GDPs) which are Hewlett-Packard (HP) computers. The code is currently being maintained on both IBM and HP equipment.

The Informix database products used are the On-Line engine (rev 5.01) and ESQL/C (rev 5.0).

The GDPs are the primary supported system for Stage II. Except for the function to calculate elapsed time, all other code is identical between the IBM and HP workstations at HRL. There are differences however in the compiling/linking instructions contained in the "make" file.

II. Directory and File Information

Each of the following sections describes the contents of the directories and important files within them. Directories are identified in two ways. The first is the full pathname on the HRL GDPs. The second, as shown in parentheses, is the .Apps_defaults token used in the software. Appendix 1 contains an explanation of the .Apps_defaults concept. Appendix 2 contains a diagram of the directory structure at HRL.

The following abbreviations apply to the file naming conventions in this section:

XXX - 3 character radar identifier
MM - 2 digit month number (01 - 12)
DD - day of month (01 - 31)
YY - 2 digit year (00 - 99)
HH - hour in Z time (00 - 23)

1. /usr/apps/precip_proc/bin (sii_bin)

This directory contains the Stage II executable and related shell scripts.

2. /usr/apps/precip_proc/scripts

This directory contains the scripts for running the StageII executable. It also contains the rerun_stageii.sh script which is executed from StageIII to rerun StageII for a list of radars and hours.

3. /usr/apps/stageii/input/param (st2_param)

This directory contains the parametric data used for Stage II processing. File names with descriptions appear below.

misbin.XXX static missing bin information for each radar -
file is binary 131x131 array of values equal to
0 or 1 where 0 = missing bin and 1 = not missing

xmptim.XXX historical mean precipitation for each HRAP
bin for use in mountainous areas - file is
binary 131x131 array (also known as PRISM data)

If the misbin array does not exist, then StageII uses an array of all 1. If the xmptim array does not exist, an array of all 1.0 is used.

4. /usr/apps/stageii/input/stagei (st2_st1 or st1_data)

This directory contains the decoded WSR-88D HDP radar products. File names are of the form XXXMMDDYYHHZ.

5. /usr/apps/stageii/input/satel (st2_satel)

In the future, this directory will contain the satellite data consisting of a 0/1 array signifying rain/no rain for use in the Ap removal process.

6. /usr/apps/stageii/logs/auto (sii_log_auto)

This directory contains the informational output from the automatic Stage II runs. This output includes a list of parameter values used for the run, a list of gage and radar values and other information concerning the run. File names are of the form OUTPUT.XXXMMDDYYHHZ.

7. /usr/apps/stageii/logs/manual (sii_log_man)

This directory contains the informational output from the manual runs. Contents of the files and file names are as above.

8. /usr/apps/stageiii/input/stageii (st3_input)

This directory contains the gage-only field and multisensor field files. These are the files which are input to the Stage III process. These files have names of the form XXXggMMDDYYHHz and XXXmlMMDDYYHHz respectively. The ml and gg files are in binary format.

Files containing the local HRAP coordinates of bins removed as part of the "Edit multisensor field" option of Stage III also appear here. They have names of the form XXXMBMMDDYYHHZ where MB stands for missing bins.

III. Informix Database Interactions

A. Table Descriptions

This section describes the Informix database tables used by Stage II.

1. S2GageRadarVal

This table contains the gage values used by Stage II for each date/hour and radar. The SIIPP process creates the records and inserts the value of the gage accumulation.

Each record in this table also contains the following information which is inserted as part of the Stage II process:

(1) local HRAP x coordinate of gage
(2) " " y " " "
(3) radar value (mm) of HRAP bin containing gage
(4) radar value (mm) which is closest in value to gage value of the 9 nearest neighbor HRAP bins
(5) minimum radar value (mm) of the 9 nearest neighbor HRAP bin values
(6) maximum radar value (mm) of the 9 nearest neighbor HRAP bin values
(7) time distribution flag
(8) gage edit flag

(9) previous gage value (default = -99.)

Note that all radar values above are NOT multiplied by a bias.
If the gage is located outside of a circle of radius 65 HRAP bins, then (3),(4),(5) and (6) above are set to -99.0. If the radar data is not available for the hour of interest, then the values are also set to -99.0. If the HRAP bin in which the gage is located is defined as missing (according to the misbin array), then the values are set to -999.0.

Hourly gage records which are the result of time distribution are designated by setting the time distribution flag to be greater than zero.

2. S3GridManip

This table stores field related information for each radar, date/hour. The number of pseudo gages, type of field (gg or ml or none) currently used to create the Stage III mosaic, the ignore radar flag for Stage II calculations, forecaster override flag and edited bias value are currently stored.

Records are initially inserted into the S3GridManip table by Stage II with the number of pseudo gages set to 0, the ignore radar flag set to 0 (i.e., do not ignore radar data), the forecaster override flag set to 0 (i.e. no override) and the edited bias value set to 0.0. The type of field is initially set to ml if a multisensor field is generated by StageII. If not, then it is set to gg if there are hourly gages available. If not, the type of field is set to blank.

Updates to the field type can be made from Stage II or Stage III. Updates to the ignore radar flag are made from StageIII only. When these updates are made from Stage III, the forecaster override flag is set to 1. The update process from Stage II first checks this flag. If set to 1, no update is made thus preventing automatic StageII runs from overriding forecaster changes to the field type.

3. HDPRadar

This table contains the information read from the HDP radar product including the supplemental data. It also contains the maximum value found in the radar field in mm. If a record for a given radar and date/hour does not appear in this table, it means that the HDP product was not decoded.

4. location

This table contains the station location information including latitude and longitude in decimal degrees.

5. PseudoGageRadarVal

This table contains the pseudo gage information. Pseudo gages are additional gages created by the forecaster running Stage III. The records include the latitude and the longitude of the gage. Other fields are the same as those found in S2GageRadarVal records except for the time distribution flag and duration which do appear as part these records.

6. radarloc

This table contains the radar location information including its latitude and longitude in decimal degrees.

7. Stage2Result

This table contains information about the Stage II runs for each radar and date/hour combination. Because the automatic runs rerun previous hours, records in this table are frequently updated. Information such as whether or not radar data was available, number of gages available and Stage II bias value are part of the records. Also included is a flag for satellite availability.

8. S2BiasCalcParams

This table contains the input data for the bias estimation. The following is a listing of each parameter followed by its default value and definition:

rng_min	**0.**	Denotes 'minimum range.' It specifies the minimum range (i.e., distance from the radar, in km) for pairing rain gage data with collocated radar rainfall data. If one opts to ignore radar rainfall estimates at close ranges, say, below 50 km (e.g., to calculate mean field biases that are not subject to the close-range bias in WSR-88D precipitation products), he/she may set **rng_min** to 50., and interpret the calculated bias accordingly: it is then an estimate of the ratio of the sum of gage rainfall amounts over the area identified as raining by both gages and the radar beyond the range of 50 km to the sum of bias-adjusted radar rainfall amounts over the same area.
rng_max	230.	Denotes 'maximum range.' It specifies the maximum range (in km) for pairing rain gage data with collocated radar rainfall data. If one opts to ignore radar rainfall estimates at far ranges, say, beyond 180 km (e.g., to calculate mean field biases that are not subject to the far-range bias in WSR-88D precipitation products), he/she may set **rng_max** to 180., and interpret the calculated bias accordingly (see **rng_min**).
nmin	7	Denotes 'minimum number.' It specifies the minimum number of valid radar-gage pairs to initiate bias calculation, i.e., there has to be at least **nmin** positive gage data and collocated positive radar rainfall data to attempt bias calculation. This implies that, loosely speaking, if there are, e.g., 35 hourly gages randomly scattered under the radar umbrella, it must rain over 20 percent of the radar umbrella to attempt bias calculation (i.e.,

35x0.2=7).

If the gage network is so sparse that bias calculation will be a rarity, one may experiment with a smaller **nmin** (say, 4 to 5): it is then no longer possible to expect that the algorithm will consistently behave rationally: for example, a sample bias estimate (i.e., the ratio of the sum of positive gage rainfall amounts to the sum of collocated positive radar rainfall amounts) based on only 4 data points may be wrongly interpreted by the algorithm as very reliable.

eps 10.$^{-6}$ Denotes ‘machine epsilon.’ The default value is much larger than the actual machine epsilon, i.e., the smallest floating point number recognized by the computer on which the algorithm is run. The purpose is to guard against zero-divide associated with numerically singular matrices. There should be no need to change this.

std_cut 2.5 Denotes ‘standard deviation cutoff.’ It is a quality control parameter (dimensionless) in pairing positive rain gage data with collocated positive radar rainfall data. It specifies the confidence interval (in units of standard deviation of the standardized error of radar rainfall data) around the linear regression through matched radar-gage pairs.

One can make a rough guess at what percentage of radar-gage pairs might get thrown out, by looking up the cumulative probably distribution table for the standard normal variate: at levels of **std_cut**=0.5, 1., 1.5, 2., and 2.5, one is throwing away, loosely speaking, 31, 16, 7, 2, and 1 percent of the data. Hence, the larger **std_cut**, the more willing one is to accept radar rainfall data in bias calculation even though they may deviate significantly from the collocated rain gage data.

The quality control step associated with **std_cut** is intended to catch apparent gross outliers, such as egregious gage data or AP-contaminated radar data, while allowing differences between radar and gage data due to natural variability of rainfall and to ‘usual’ errors associated with radar observation of rainfall. Setting **std_cut** to a smaller value (e.g., 1.5) will produce biases that are less variable, but will almost certainly deteriorate mass balancing between bias-adjusted radar rainfall data and rain gage data.

alpha 100. Denotes the memory span in the updating of state variables in the algorithm. Loosely speaking, it specifies the length of the time window (in hours) in blocking out radar-gage pairs to be used in bias calculation. Loosely speaking again, if **alpha** is set to 10000./1000./100./10./1., the calculated bias is based on radar-gage pairs collected from the most recent 10,000/1,000/100/10/1 hours (≈13.5 months/1.3 months/4 days/10 hours/1 hour). If **alpha** is less than 1., bias is calculated using essentially radar-gage pairs from the current hour only.

In actuality, the algorithm employs a time window which is not as clear-cut as the one described above, but which fades out as the age (i.e., the time of bias calculation - the observation time) of the radar-gage pair increases. The purpose of this fuzzy time window is to accommodate recursive estimation, in which case there is no need to actually store hundreds or thousands of radar-gage pairs: all the algorithm needs is several state variables from the most recent bias calculation.

One is strongly urged to play with this parameter to best tune the algorithm to the local gage network density (and, to a lesser extent, to the local rainfall climatology). A large **alpha** reduces timeliness of the bias estimate, but increases the chance of getting a bias calculated as well as stability in the calculated bias (i.e., no very big or small biases).

z_cut	0.01	Denotes ‘observed rainfall cutoff.’ It specifies the smallest gage or radar rainfall amount to be included in the bias calculation (in mm). If, for whatever reason, one is interested only in estimating bias for heavy rainfall situations, he/she may increase the **z_cut** setting to throw out small rainfall amounts: the consequence is that 1) it will greatly deteriorate radar umbrella-wide mass balancing between bias-adjusted radar rainfall and rain gage rainfall and 2) bias calculation will occur far less frequently.
cv_cut	0.33	Denotes ‘coefficient of variation cutoff.’ It is a quality control parameter (dimensionless) to check if the calculated bias is acceptable. To measure how certain/uncertain the calculated bias is, the algorithm calculates the coefficient of variation of the bias calculated (≡error standard deviation of the calculated bias/calculated bias itself). Too large a coefficient of variation is an indication that, even though a bias is calculated, it is not trustworthy. If, for example, the calculated bias and error standard deviation is 1.5 and 0.2, respectively, it means that there is approximately 84 and 98 percent chance that the true (unknown) bias lies within 1.5±0.2 and 1.5±0.4, respectively (this interpretation is an extremely loose and liberal one because bias is not distributed normally, but lognormally). Hence, a larger/smaller value of **cv_cut** implies that one is more/less willing to accept an uncertain bias estimate.
aver_cut	0.5	Denotes ‘average radar rainfall cutoff.’ It is a quality control parameter (in mm) to avoid bias calculation in light rainfall situations.

Because of nonlinear errors in radar rainfall data (due, e.g., to inaccurate Z-R relationships, range degradation, etc.), bias is rainfall magnitude-dependent: if there are lots of radar-gage pairs so that one could calculate biases over various ranges of rainfall amounts, the biases will not be the same, but vary greatly depending on the range of rainfall amount (typically, biases are larger/smaller for small/large rainfall amounts).

If there is only light rainfall in the radar umbrella, it is very likely that the sample bias (≡ sum of gage rainfall/sum of collocated radar rainfall) will be very high simply because the denominator is very small. To prevent this from adversely affecting the bias calculation, mean radar rainfall, conditional on occurrence of rainfall, is calculated and compared with **aver_cut** before bias calculation is initiated. If **aver_cut** is greater, it is interpreted that it is raining too lightly for the sample bias to mean much (i.e., the signal-to-noise ratio is too small), and no bias calculation is attempted.

Applying the above observation, one can actually foresee how the calculated bias might behave over the course of the life cycle of a storm (assuming, for the sake of argument, that the storm remains stationary from birth to death): the cycle of birth-development-maturing-dissipation-death would roughly translate into the bias cycle of high-medium/small-small-small/medium-high. Apparently unrealistically high biases are most likely an indication that it is raining very lightly (and hence not much rain water to hydrologically worry about).

9. S2GeneralParams

This table contains general input parameters. The following is a listing of each parameter followed by its default value and definition:

process_type 1
Type of StageII processing
1 = new gg/ml field generation methods
0 = old gg/ml field generation methods

ap_removal_test 0
on/off flag for Ap removal test
there is currently no Ap removal algorithm in StageII
0 = off

outlier_thresh 500.
Maximum radar data value allowed in mm

gg_ml_compute_var 0
Variance compute flag in gg and ml field calculations

0 = off

1 = on

compute_time_dist 0
on/off flag for time distribution calculations
there is currently no time distribution algorithm in StageII
0 = off

max_gage_dur 3
maximum gage duration for time distribution

max_num_gage 10
maximum number of gages to time distribute

10. S2ggGridParams

This table contains the input data for the gage only field calculation. The following is a listing of each parameter followed by its default value and definition:

distmin 0.1 Specifies the separation distance in km in correlation modeling. No changes are recommended.

itype 2 Specifies the type of gage-only analysis procedure to be used (**itype**=1: Reciprocal Distance-Squared method (RDS), **itype**=2: Single Optimal Estimation (SOE), **itype**=3: Double Optimal Estimation (DOE)). Computationally, RDS is the least expensive, DOE the most expensive.

If there are less than 50 gages or so under the radar umbrella (assuming that they are fairly well-scattered), RDS is recommended. If there are more gages, SOE is recommended. DOE, developed primarily to assimilate satellite and lightning data in the future, is twice as computationally expensive as SOE, and hence is not recommended at this time.

nbors 4 Specifies the number of surrounding rain gage measurements to be used in the estimation procedure (if that many gages indeed exist within the radius of **radius** (see below)). The maximum is 20.

This parameter has a huge impact on both the accuracy of estimates and how the estimated rainfall field looks like. If **nbors=1**, the estimate will be the same as the nearest rain gage datum (if there exists at least a single rain gage within the radius of **radius**): it will produce something that looks nothing like a rainfall field. Theoretically, the larger **nbors** is, the more accurate the estimates are (if that many rain gages indeed exist within the radius of **radius**).

Computationally, however, the larger **nbors**, the more expensive. With the recommended value of 4, this trade-off between accuracy and computational burden should not be an issue at most sites because of the sparsity of rain gage networks.

rainmin	0.01	Specifies the minimum detectable rainfall depth in mm. No changes are recommended.
rangei	52.	Specifies the indicator correlation scale in km. It represents a characteristic scale for spatial intermittency of rainfall. The default value is obtained from climatological analyses of WSR-88D hourly rainfall data in the Southern Plains. Until site- and seasonality-specific estimates of **rangei** can be obtained, no changes are recommended.
cor0i	1.	Specifies the lag-0 indicator correlation coefficient. No changes are recommended.
cor0pi	0.75.	Specifies the lag-0+ indicator correlation coefficient. No changes are recommended.
rangec	36.	Specifies the conditional correlation scale in km. It represents a characteristic scale for within-storm variability of rainfall. The default value is obtained from climatological analyses of WSR-88D hourly rainfall data in the Southern Plains. Until site- and seasonality-specific estimates of **rangec** can be obtained, no changes are recommended.
cor0c	1.	Specifies the lag-0 conditional correlation coefficient. No changes are recommended.
cor0pc	1.	Specifies the lag-0+ conditional correlation coefficient. No changes are recommended.
radius	52.	Specifies the radius of influence in km. For example, with the default value of 52, rain gage data outside of the circle of radius of 52 km are not used in estimation. Theoretically speaking, **radius** should always be equal to max{**rangei**,**rangec**}. However, because we do not have site- and seasonality-specific estimates of **rangei** and **rangec** yet, it is recommended that **radius** be adjusted to account for differences between, e.g., cellular/convective and widespread/stratiform rainfall fields. Generally speaking, **radius** for a convective storm should be smaller than

for a stratiform storm.

It is recommended that the sensitivity of the analysis field on **radius** be familiarized by displaying gage-only analysis fields for a range of values of **radius** (say, 20 to 100 km) so that a visual ‘feel’ may be acquired for what the appropriate values of **radius** should be for the regional and seasonal climatology of rainfall (and, ideally, synoptic conditions as well).

11. S2mlGridParams

This table contains the input data for the multisensor field calculation. The following is a listing of each parameter followed by its default value and definition:

itype 1 Specifies the type of multi-sensor estimation procedure (**itype**=1: Single Optimal Estimation (SOE), **itype**=2: Double Optimal Estimation (DOE)).

DOE, developed primarily to assimilate satellite and lightning data in the future, is computationally twice as expensive as SOE, and hence is not recommended at this time.

nbors 3 Specifies the number of the nearest gage measurements to be used in estimation (if that many gages indeed exist within the radius of influence: in multi-sensor estimation, the radius of influence is computed in real time from radar rainfall data).

Theoretically speaking, the larger the value of **nbors**, the more accurate the estimates will be. As in gage-only estimation, however, the larger the value of **nbors** (if that many gages indeed exist within the radius of influence), the more computationally expensive. Changing **nbors** does not have as great an impact in multi-sensor estimation as it does in gage-only estimation. The current default of 3 (if that many gages indeed exist within the radius of influence) is strongly recommended unless excessive CPU/elapsed time becomes a problem, in which case **nbors** may be lowered to 2 (the procedure will still produce reasonable-looking rainfall fields even with **nbors**=1).

distmin 0.1 Specifies the separation distance in correlation model in km. No changes are recommended.

rainmin 0.01 Specifies the minimum detectable rainfall depth in mm. No changes are recommended.

radius 52. Specifies the radius of influence in km. See explanation under S2ggGridParams table

crscori 0.80 Specifies the lag-0 indicator cross-correlation coefficient. Loosely speaking, this quantifies, on the scale of 0 to 1, how accurate radar rainfall data are, on the average, in discerning rainfall/no-rainfall from no-rainfall/rainfall.

The default value is obtained from climatological analyses of hourly rain gage data and WSR-88D radar rainfall data in the Southern Plains. Data analysis-permitting, **crscori** should be estimated for each site as a function of range and seasonality (ideally, it should also be stratified according to storm type; e.g., convective versus stratiform). In the meantime, no changes are recommended.

crscorc 0.85 Specifies the lag-0 conditional cross-correlation coefficient. Loosely speaking, this quantifies, on the scale of 0 to 1, how accurate radar rainfall data are, on the average, in estimating rainfall amount over rain area (i.e., given that it is raining).

The default value is obtained from climatological analyses of hourly rain gage data and WSR-88D radar rainfall data in the Southern Plains. Data analysis-permitting, **crscorc** should be estimated for each site as a function of range and seasonality (ideally, it should also be stratified according to storm type; e.g., convective versus stratiform). In the meantime, no changes are recommended.

scali_def 36. Specifies the default indicator correlation scale in km. It is used whenever the indicator correlation scale (the same as **rangei** in gage-only estimation) cannot be obtained in real time. Until site- and seasonality-specific estimation of the parameter, no changes are recommended.

scale_def 28. Specifies the default conditional correlation scale in km. It is used whenever the conditional correlation scale (the same as **rangec** in gage-only estimation) cannot be obtained in real time. Until site- and seasonality-specific estimation of the parameter, no changes are recommended.

scali_max 70. Specifies the maximum indicator correlation scale in km. It is the upper bound for the indicator correlation scale. No changes are recommended.

scale_max 50. Specifies the maximum conditional correlation scale in km. It is the upper bound for the conditional correlation scale. No changes are recommended.

threshmin 0.25 Specifies the minimum threshold value in mm for multi-sensor estimates (display purposes only). No changes are recommended.

IV. Executing Stage II

A. Automatic Mode

Stage II runs ending at the current date and hour are referred to as automatic mode runs. In automatic mode, the radar identifier and number of hours to run are passed to Stage II from the command line. For example,

```
/usr/apps/stageii/bin/StageII  TLX  5
```

runs Stage II for the TLX radar for the current hour plus the previous 4 hours. The current date and hour for the run are read from the system clock. The output from automatic runs are sent to /usr/apps/stageii/logs/auto (st2_log_auto).

At RFCs running StageII, automatic mode runs are executed through the auto_stageii script submitted via the crontab. At OH, this script is submitted once per hour on the half hour. This choice for submittal is arbitrary and auto_stageii can be submitted more frequently if warranted.

B. Manual Mode

Stage II runs ending at any date and hour are referred to as manual mode runs. In manual mode, the radar identifier, number of hours to run, ending hour and ending date (mmddyy) in that order are passed to Stage II from the command line. For example,

```
/usr/apps/stageii/bin/StageII  TLX  5  12  030294
```

runs Stage II for the TLX radar for the 5 hours ending at 12z on 3/2/94. The output from these runs are sent to /usr/apps/stageii/logs/manual (st2_log_man).

Currently, 24 hours is the maximum length for a manual run.

C. Rerun from Stage III

Stage II reruns from Stage III are manual mode runs with the number of hours back set to 1 and the radar identifier, ending hour and ending date passed to Stage II through the .rerunii file in the user's home directory. The .rerunii file is generated automatically when changes are made in Stage III and is deleted once Stage II has been rerun.

D. Input and Output

Input to Stage II consists of the hourly digital precipitation (HDP) product from the WSR-88D radar, the Stage I radar bias (which is included in the HDP product header), hourly gage values in mm and static data from /usr/apps/stageii/input/param (st2_param). The HDP products and the gage data are transmitted to the RFCs via AFOS.

Output from Stage II which is input to Stage III consists of the gage-only and multisensor fields.

Other output is written to the .../logs/auto or .../logs/manual directories. This output contains a list of the gage and radar values and information on the performance of the algorithms. Other information is written to the Informix Stage2Result and S3GridManip tables.

V. Algorithm Descriptions

The following sections briefly describe the five Stage II algorithms. A full explanation of each appears in the document "Stage II Precipitation Processing" originally written by Dr. James Smith and updated by Rob Shedd and Richard Fulton.

A. Quality Control

This algorithm consists of a radar data outlier removal process and an anomalous propagation (AP) removal process. The outlier removal process removes radar values which are larger than a specified value and replaces it with the average of the values from the 8 surrounding HRAP bins. The AP removal process will compare the rain/no rain satellite data with the radar data to remove areas of radar data deemed to be contaminated by anomalous propagation.

B. Time Distribution

Time distribution is the method for changing multiple hour duration gage observations into hourly gage values for use by StageII. For example, a 3 hour duration gage value with obs time of 16z, will generate hourly gage values with "obs times" of 14z, 15z, 16z. Recall that the obs time is the END time of the period.

The maximum duration for time distribution will be an adaptable parameter. WFOs need to time distribute 2,3,4,5,6 hour gage reports. RFCs need to time distribute only up to 5 hour reports because the Post Analysis process handles the time distribution of 6 hour reports.

The following information outlines the current thoughts about the time distribution algorithm. It has NOT been implemented as part of StageII.

Define MAX_RADAR_RANGE parameter = maximum distance in HRAP bins a radar can be to a gage (in the winter season, this distance may be reduced)

Determine which radar the gage is closest to and within a distance of MAX_RADAR_RANGE

```
If data exists for all hours of time dist for this radar, then

   GG(h) = multiple hour value ending at hour h

   G(h) = resulting hourly value at hour h

   A(h) = weighting factor for hour h
        = radar value for hour h at the gage location divided by
          sum of radar values for all hours of time dist at gage
          location

   G(h) = A(h) * GG(h)

Else If radar data exists for all hours of time dist for another
  radar which the gage is under and that is within a distance of
  MAX_RADAR_RANGE then follow steps above

Else

   do not time distribute the gage value

End if
```

Pseudo gages created in StageIII are not included in the time distribution of a gage.

Time distributed gages are NOT included in the bias adjustment procedure.

A multiple hourly gage with value = 0.0, is time distributed without going through any of the above calculations. Hourly gage values of 0.0 are generated for all hours of the time distribution.

In the future, the following logic will be added:

- if radar data cannot be used to time distribute a gage OR if the gage is in an area determined to be blocked (using misbin array), then use gages within a given radius (defined by a StageII input parameter) to time distribute

- use an inverse distance weighting scheme to determine given to surrounding gages

C. Gage-Only Field Generation

This algorithm combines hourly rain gage values to generate precipitation estimates on an HRAP grid. In the old implementation, at each grid point, gage values are weighted as an exponential function of distance to the gage. Radar data is used to shape the field. The new method uses an optimal estimation procedure.

D. Multisensor Field Generation

This algorithm calculates and applies a new radar bias based on rainfall data from the current, previous and future hours. Because the bias calculation involves data from previous as well as future hours, a value greater than 1.0 may be calculated for the current hour even if there are no gages reporting for the current hour. The gage-only field is then merged with the adjusted radar data to generate the multisensor field. In the old method, this merging is done by calculating a weighting factor for the gage only field value and StageII adjusted value at each HRAP bin. The new method uses an optimal estimation procedure.

E. Bias Estimation

The full technical description of the bias estimation algorithm will be provided in July 1997 as a part of the OH's Final Report to the NEXRAD Program and OSF.

Conceptually, the algorithm is extremely simple (no matter how complicated the technical description may appear in the aforementioned soon-to-be-available Final Report). In a nutshell, the bias at the current hour is given by the ratio of the sum of all positive gage rainfall data over the radar umbrella (this is the spatial window of sampling) from the previous x number of hours (this is the temporal window of sampling) to the sum of all positive HDP rainfall data at the same gage locations over the same spatio-temporal window of sampling.

In actuality, the temporal window is not as clear-cut as described above (one side of the window is actually kind of fuzzy) in order to accommodate recursive estimation (let's just say that this results in tremendous savings in CPU and RAM).

The size of the temporal window, x, is specified by the adaptable parameter 'mem_span' (in hours). If you set 'mem_span' to 1, you are using radar-gage pairs only from the current hour (i.e., you are calculating the sample bias). If you set it 720, you are using radar-gage pairs from the most recent 30 days (i.e., you are calculating the monthly bias). If you set ‘mem_span’ to 24*365, you are calculating the yearly bias. If you set it to an extremely large number, you are calculating the climatological bias. The optimal choice for ‘mem_span’ depends largely on the gage network density under the radar umbrella. Qualitatively speaking, the denser the network is, the smaller ‘mem_span’ should be to capture the temporal variability of the bias. Very often, however, one is much better off gaining reliability in the bias estimate by setting ‘mem_span’ to a larger value, thereby collecting more data (some initial guidelines are given below) than being defeated by sampling errors due to lack of data while futilely trying to capture the temporal variability of the bias. In other words, there is a trade-off between reliability and timeliness (they are mutually exclusive in bias estimation).

The initially recommended value for ‘mem_span’ is 100 or larger for most sites. For those enviable gage-rich sites in OHRFC and ABRFC, ‘mem_span’ may be set to 50 or even less. We strongly recommend that you tune ‘mem_span’ until you reach the setting to you liking. It is important to reiterate, however, that ‘mem_span’ is not some dimensionless free parameter that must be estimated through some fancy optimization process, but has a physical meaning which allows you to make an educated guess at what you may be getting in your bias. For example, suppose a WSR-88D umbrella has 50 gages. Suppose that it has been raining for 10 hours over 20 percent of the radar umbrella. If your ‘mem_span’ is set to 10, then at the end of this event what you are getting is essentially the ‘storm’ bias based on roughly 50*10*0.2=100 radar-gage pairs. This bias is then used as the initial bias for the next storm.

The size of the spatial window is specified by adaptable parameters ‘min_gage_rad_dist’ and ‘max_gage_rad_dist’ (both in kilometers). For example, if you set ‘min_gage_rad_dist’ and ‘max_gage_rad_dist’ to 0 and 230, respectively, you are collecting radar-gage pairs for bias calculation from the entire radar umbrella. Initially, we do not recommend any changes to these parameters.

Because of range-dependent biases in HDP products, it is expected that bias-adjusted HDP estimates based on ‘min_gage_rad_dist’ and ‘max_gage_rad_dist’ settings of 0 and 230, respectively, may be overestimates over mid-ranges. If this becomes a problem, one may set ‘max_gage_rad_dist’ to a smaller value (say, 200 km) to ignore radar rainfall data suffering from far-range degradation. The downside of this is of course that you will be losing one-fourth of the data (yes, the thin annulus between the range rings 200 and 230 km actually amounts to almost a quarter of the whole radar umbrella).

Off-line studies using long-term data (to be available in the Final Report) indicate that the algorithm does a better job at large-scale mass-balancing. In other words, if you sum up a year’s worth of gage rainfall amounts collected over the radar umbrella and corresponding bias-adjusted radar rainfall amounts, they will be approximately the same (say, within + or - 10 percent). This implies that bias-adjusted radar rainfall amounts are equally likely to be overestimates as they are underestimates.

Another consequence of mass-balancing bias adjustment is that small-scale accuracy of bias-adjusted radar rainfall may be worse than that of raw HDP rainfall. In other words, if you spot a small area of intense rainfall, you may find that raw HDP estimates are more accurate that the bias-adjusted estimates. This catch-22 problem (in that you cannot always reduce mean error and root mean square error at the same time) is due to rainfall amount-dependent errors in HDP products stemming largely from range degradation and inaccurate Z-R parameters. At any rate, beware that bias-adjusted radar rainfall may be overestimates in cores of heavy rainfall!

VI. Other Topics

A. Missing Bins

The missing bin data refers to HRAP bins in a 131x131 array for each radar which are either known to be consistently bad or have been removed for a particular hour by a forecaster using the "edit multisensor field" feature of Stage III. Each array element equals 0 or 1 corresponding to missing or non-missing.

The missing bin data consists of static and dynamic components. The static component represents areas under each radar which are always blocked by mountains or tall buildings and therefore determined to be consistently bad. The dynamic component represents areas deemed by the forecaster running Stage III to have bad radar data or other problems for a single hour.

Currently, Stage III passes to Stage II the coordinates of all HRAP bins which are to be removed in a flat file. Stage II checks for the existence of these dynamic files for each hour and for each radar. If such a file is found, Stage II creates a new missing bin array combining the static and dynamic components.

Missing bin data is used by Stage II in the bias computation, summary statistics calculations and the gage-only and multi-sensor field generation.

Appendix 1 - .Apps_defaults

The apps_defaults system was originally designed to eliminate the need for direct references to directories from source code and scripts. It has evolved into a general purpose system for string substitution into a token. The "value" of the token is found by searching for it in a user created file or from the user's environment variable list. The get_apps_defaults function performs this look up. The file containing the tokens to be searched is called .Apps_defaults.

The search for a token is done in the following order:

    environment variable
    user's .Apps_defaults file
    system level .Apps_defaults file

The filename of the user's .Apps_defaults file is defined by the environment variable APPS_DEFAULTS_USER. The system level file is defined by APPS_DEFAULTS. Both environment variables may be set to point to the same file.

The syntax used in .Apps_defaults files is:

```
  token : resource
```

where: token is defined as a string delimited by white space or the delimiter
the delimiter between token and resource is the :

The following rules apply to .Apps_defaults files:

- comments are indicated by a #
- tokens and resources cannot begin with a # or :
- blank lines are allowed
- referbacks are indicated by $(...) where '...' is resolved in the same manner as any other token. A substitution is made for the $(...) string to create the final resource value. Multiple referbacks are allowed in a resource, but embedded referbacks are not allowed i.e. $($(...)) is not allowed.

The following text is taken from an .Apps_defaults file at HRL:

```
stgii_dir         : /usr/apps/stageii          # Stage II
                                               #  top-level dir

st2_input         : $(stgii_dir)/input
```

```
st2_param          : $(st2_input)/param       # stageII parameter
                                              #  values dir
st2_st1            : $(st2_input)/stagei      # stageI data dir
st1_data           : $(st2_input)/stagei      # stageI data dir
                                              #  used by StageIII
st2_satel          : $(st2_input)/satel       # satellite data dir

st2_log            : $(stgii_dir)/logs        # logs top level dir
st2_log_auto       : $(st2_log)/auto          # logs for auto runs
st2_log_man        : $(st2_log)/manual        # logs for manual
                                              #  runs

st3_dir         : /usr/apps/stageiii       # main stage 3 dir
st3_data        : $(st3_dir)/input         # main stage 3 data
                                           #  directory
st3_input       : $(st3_data)/stageii      # dir containing
                                           #  gg,ml files
```

Appendix 2 - Directory Structure

```
/usr/apps/stageii
                 /bin
                 /input
                        /param
                        /satel
                        /stagei
                 /logs
                        /auto
                        /manual
```